



Automatic motor task selection via a bandit algorithm for a brain-controlled button

Joan Fruitet, Alexandra Carpentier, Rémi Munos, Maureen Clerc

► To cite this version:

Joan Fruitet, Alexandra Carpentier, Rémi Munos, Maureen Clerc. Automatic motor task selection via a bandit algorithm for a brain-controlled button. [Rapport de recherche] RR-7721, INRIA. 2011. inria-00624686

HAL Id: inria-00624686

<https://hal.inria.fr/inria-00624686>

Submitted on 20 Sep 2011

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

***Automatic motor task selection via a bandit
algorithm for a brain-controlled button***

Joan Fruitet — Alexandra Carpentier — Rémi Munos — Maureen Clerc

N° 7721

Septembre 2011

Domaines 1,5

 ***rapport
de recherche***

Automatic motor task selection via a bandit algorithm for a brain-controlled button

Joan Fruitet^{*}, Alexandra Carpentier[†], Rémi Munos[†], Maureen Clerc^{*}

Domaines : Mathématiques appliquées, calcul et simulation
STIC pour les sciences de la vie et de l'environnement
Équipes-Projets Athena et SequeL

Rapport de recherche n° 7721 — Septembre 2011 — 14 pages

Abstract: This study presents a new procedure to automatically select a discriminant motor task for an asynchronous brain-controlled button. This type of control pertains to Brain Computer Interfaces (BCI). When using sensorimotor rhythms in a BCI, several motor tasks, such as moving the right or left hand, the feet or the tongue, can be considered as candidates for the control. This report presents a method to select as fast as possible the most promising task. We develop for this purpose an adaptive algorithm *UCB-classif* based on the stochastic bandit theory and build an EEG experiment to test our method. By not wasting time on inefficient tasks, our algorithm can focus on the most promising ones, resulting in a faster task selection and a more efficient use of the BCI training session. This leads to better classification rates for a fixed time budget, compared to a standard task selection.

Key-words: Brain Computer Interfaces, bandit algorithm, motor imagination, sensorimotor rhythms, ERS, ERD, task selection

^{*} Athena project team, INRIA, Sophia Antipolis, France

[†] SequeL project team, INRIA, Lille, France

Sélection automatique de tâches motrices par un algorithme de bandit pour le contrôle d'un bouton par la pensée

Résumé : Cette étude présente une nouvelle procédure pour sélectionner automatiquement une tâche motrice discriminante pour contrôler un bouton par la pensée. Ce type de contrôle relève du domaine des interfaces cerveau-ordinateur, ou Brain Computer Interface (BCI). Dans les BCI basées sur les rythmes sensorimoteurs cérébraux, différentes tâches motrices peuvent être considérées, comme le mouvement de la main droite ou gauche, des pieds ou de la langue.

Ce rapport présente une méthode afin de sélectionner le plus rapidement possible la tâche la plus prometteuse. Nous avons développé à cet effet un algorithme adaptatif *UCB-classif* basé sur la théorie bandit stochastique, et créé une nouvelle expérience EEG pour tester notre méthode. Cet algorithme évite de perdre du temps sur des tâches inefficaces, ce qui permet une sélection plus rapide et une utilisation plus efficace de la session d'apprentissage. Cela conduit à de meilleurs taux de classification pour un budget de temps fixé, par rapport à une sélection de tâche standard.

Mots-clés : Interfaces Cerveau Ordinateur, algorithm de bandits, rythmes sensorimoteur, sélection de tâches

1 Introduction

Scalp recorded electroencephalography (EEG) can be used for non-muscular control and communication systems, commonly called brain-computer interfaces (BCI). BCI systems based on sensorimotor rhythms (SMR) rely on the users' ability to control their SMR in the mu (8-13Hz) and/or beta (16-24Hz) frequency bands [1, 2, 3]. Indeed, these rhythms are naturally modulated during real and imagined motor action.

More precisely, real and imagined movement similarly activate neural structures located in the sensorimotor cortex, which can be detected in EEG recordings as synchronisation (event related synchronisation or ERS) and/or desynchronisation (event related desynchronisation or ERD) in the mu and beta frequency bands [4, 5]. Because of the homuncular organisation of the sensorimotor cortex [6], it is possible to distinguish different limb movements according to the position of the neural structures involved. For example, a right hand movement involves a modification of cortical activity on the upper left precentral gyrus, whereas the modification of activity due to feet movement is less lateralized.

BCI based on the control of sensorimotor rhythms generally use movements lasting several seconds, that enable continuous control of multidimensional interfaces [1]. On the contrary our work targets a brain-controlled button that can be rapidly triggered by a short motor task [7, 2].

A vast variety of motor tasks can be used in this context, like imagining rapidly or slowly moving the hand, grasping an object or kicking an imaginary ball. These different tasks permit different levels of control. Unfortunately, the task which enables the best control of a BCI varies for each user [8]. Generally a preliminary training session is done to determine which motor task, among a small subset of tasks, is the most appropriate for each user [9, 10, 11]. Although task selection is often mentioned as a preliminary step in the BCI literature, this article is the first, to our knowledge, to propose a principled method for this purpose.

This study presents a way of automatically and efficiently evaluating and selecting, for a given user, the most appropriate motor task. The benefit is twofold: reducing the length of the training stage, and exploring a larger variety of motor tasks.

We thus developed an adaptive algorithm, which, compared to a standard selection method, is capable of rapidly eliminating non-efficient tasks in order to focus on the most promising ones. The task selection procedure is based on bandit theory (initiated in [12]), and its goal is to rapidly select an action that maximizes the expected performance given a limited budget of trials.

The rest of the article is organized as follows: in Section 2, we describe the EEG experiment we built to simulate the on-line use of a brain-controlled button, and we model the task selection as an optimization problem, which is solved using an Upper Confidence Bounds algorithm. We motivate the choice of this algorithm by proving its theoretical performance. Section 3 presents results on simulated on-line experiments, Section 4 discusses these results and presents perspectives, while conclusions are drawn in Section 5.

2 Material and Methods

2.1 The EEG experiment

The experiment was designed to be as close as possible to the online use of a brain-controlled button. To this aim, we presented, at random timing, cue images during which the subjects were supposed to perform 2 second long motor tasks (intended to activate the button). During an online use of our adaptive algorithm, the order of the presentations would be determined in view of selecting most rapidly the best motor task.

Six right-handed subjects, ages 24 to 39, with no disabilities, were sitting at 1.5m of a 23' LCD screen. Scalp electrodes were recorded through an Open-ViBE platform [13] at a sampling rate of 512Hz, on 11 out of 64 channels of a TMSI amplifier (see Figure 1). The signal was band-pass filtered and a spatial Laplacian was applied to increase the signal to noise ratio.

The experiment was composed of 5 to 12 blocks of approximately 5 minutes. During each block, 4 cue images were presented for 2 seconds in a random order, 10 times each. The time between two image presentations varied between 1.5s and 10s. Each cue image was a prompt for the subject to perform the corresponding motor action during 2 seconds, namely moving the right or left hand, the feet or the tongue. Otherwise the subjects stayed as motionless as possible.

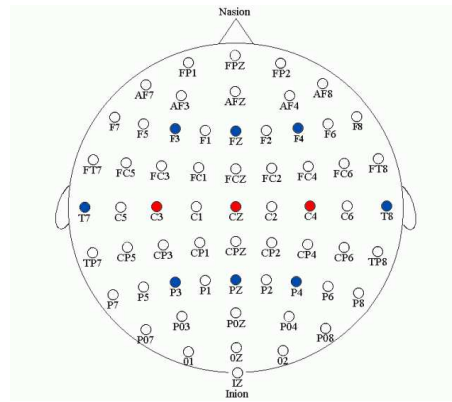


Figure 1: The 64 EEG cap with the 3 electrodes from which the features are extracted (in red). The additional electrodes used for the Laplacian are in blue.

2.2 Feature extraction

In the case of short motor tasks, the movement (real or imagined) produces an ERD in the mu and beta bands during the task, and is followed by a strong ERS [7] (sometimes called beta rebound as it is most easily seen in the beta frequency band).

We use both the ERD during the movement and the ERS following the movement to detect real or imagined movements. Although using the rebound (or ERS that follows the movement) may not be practical for BCI, because it increases the latency of the response, we believe that the rebound may help

the BCI user in early stages, by enhancing the classification rate, and it could subsequently be discarded when the subject is sufficiently trained.

We extracted features of the mu and beta bands during the 2-second time windows of the motor action and on a subsequent 1.5 second of signal in order to use the bursts of mu and beta power (ERS or rebound) that follow the potential movement. Figure 2 shows a time-frequency map on which the movement and rebound windows are superimposed. We can see that during the movement the power in the mu and beta bands decreases (ERS) and that approximately 1 second after the movement it increases to reach a higher level than in the resting state (ERS).

More precisely, the features were chosen as the power around 12Hz and 20Hz extracted at 3 electrodes over the sensorimotor cortex (C3, C4 and Cz). Thus, 6 features are extracted during the movement and 6 during the rebound. The length and position of the windows and the frequency bands were chosen according to a preliminary study with the first subject and were deliberately not tuned for the other subjects.

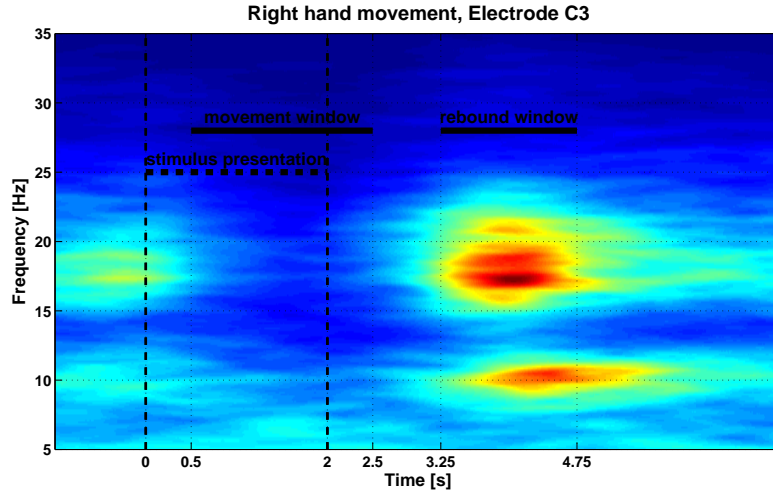


Figure 2: Time-frequency map of the signal recorded on electrode C3, after spatial Laplacian, for a right hand movement (subject 1).

2.3 Modeling the problem

Let K denote the number of different images (we will choose $K = 4$) and N the total number of images (the budget) that will be presented to the subject during the learning stage. Our goal is to find a selection strategy (i.e. that chooses at each time-step $t \in \{1, \dots, N\}$ an image $k_t \in \{1, \dots, K\}$ to present) which enables to select the most discriminative movement (i.e. with highest classification rate in generalization). Note that, in order to learn an efficient classifier, we need as much training data as possible, so our selection strategy should rapidly select the most promising images in order to obtain more samples from these rather than from the others.

This issue is relatively close to the *stochastic bandit problem* [12, 14]. The classical *stochastic bandit problem* is defined by a set of K actions (pulling different arms of bandit machines) and to each action is assigned a reward distribution, which is initially unknown from the learner. At time $t \in \{1, \dots, N\}$, if we choose an action $k_t \in \{1, \dots, K\}$, we receive a reward sample drawn independently from the distribution of the corresponding action k_t . The goal is to find a selection strategy which maximizes the sum of obtained rewards.

We model the K different images to be displayed as the K possible actions, and we define the reward as the classification rate of the corresponding motor action. In the *bandit problem*, pulling a bandit arm directly gives a stochastic reward which is used to estimate the distribution of this arm. In our case, when we display a new image, we obtain a new data sample for the selected motor action, which provides one more data sample to train or test the corresponding classifier and thus obtain a more accurate performance. The main difference is that for the *stochastic bandit problem*, the goal is to maximize the sum of obtained rewards, whereas ours is to maximize the performance of the final classifier. However, the strategies are similar: since the distributions are initially unknown, one should first explore all the actions (exploration phase) but then rapidly select the best one (exploitation phase). This is called the *exploration-exploitation trade-off*.

2.4 The UCB-classif algorithm

The image selection strategy is designed by using a variant of the Upper Confidence Bound (UCB) algorithm [14], which builds high probability upper confidence bound on the mean reward value of each action, and selects at each time step the action with highest bound.

The upper bounds $B_{k,t}$ (of action k at time t) are defined in the *stochastic bandit problem* as the sum of the empirical mean of the rewards obtained for action k and a confidence interval term which depends on the number of times $T_{k,t}$ action k has been chosen up to time t :

$$B_{k,t} = \hat{r}_{k,t} + \sqrt{\frac{a \log N}{T_{k,t-1}}}, \quad (1)$$

where $a > 0$ is a constant.

We adapt the idea of this UCB algorithm to our adaptive classification problem and call this algorithm *UCB-classif* (see the pseudo-code in Table 1). The algorithm builds the $B_{k,t}$ -values (1) where $\hat{r}_{k,t}$ now represents an estimation of the classification rate built from a q -fold cross-validation technique. The cross-validation uses a linear SVM classifier based on the $T_{k,t}$ data samples obtained (at time t) from movement k . Writing r_k^* the classification rate for the optimal linear SVM classifier (which would be obtained by using a infinite number of samples), we have the property that $B_{k,t}$ is a high probability upper bound on r_k^* : the probability $p(B_{k,t} < r_k^*)$ decreases to zero polynomially fast (with N). The constant a is a measure of complexity (VC-dimension) of the class of linear SVM classifiers.

Figure 3 illustrates how the *UCB-classif* algorithm works. The intuition behind the algorithm is that it selects at time t an action k_t either because it has a good classification rate $\hat{r}_{k,t}$ (thus it is interesting to obtain more samples from

The UCB-Classif Algorithm**Parameters:** a, N, q Present each image q times (thus set $T_{k,qK} = q$).**for** $t = qK + 1, \dots, N$ **do** Evaluate by a q -split Cross Validation the performance $\hat{r}_{k,t}$ of each action. Compute the UCB: $B_{k,t} = \hat{r}_{k,t} + \sqrt{\frac{a \log N}{T_{k,t-1}}}$ for each action $1 \leq k \leq K$. Select the image to present: $k_t = \arg \max_{k \in \{1, \dots, K\}} B_{k,t}$. Update T : $T_{k_t,t} = T_{k_t,t-1} + 1$ and $\forall k \neq k_t, T_{k,t} = T_{k,t-1}$ **end for**

Table 1: Pseudo-code of the UCB-classif algorithm.

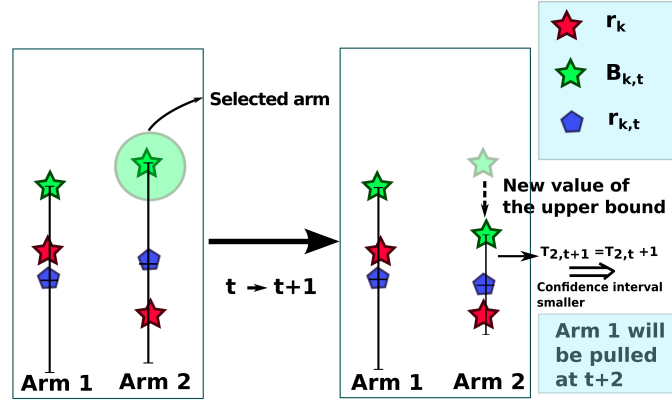


Figure 3: This figure represents two snapshots, a time t and $t + 1$, of a bandit with 2 arms. Although arm 1 is the best arm ($r_1^* > r_2^*$, represented by the red stars), at time t , $B_{1,t} < B_{2,t}$, therefore the arm 2 is selected. Pulling the arm 2 gives a better estimate $\hat{r}_{2,t+1}$ of r_2^* and reduces the confidence interval. At time $t + 1$, $B_{1,t+1}$ will be greater than $B_{2,t+1}$, so arm 1 will be selected.

it, to perform exploitation) or because its classification rate is highly uncertain since it has not been sampled many times, i.e., $T_{k,t}$ is small (thus it is important to explore it more). This enables to sample more often the action that has the highest classification rate. It is indeed important to sample the best action as much as possible in order to build the best possible classifier. The *UCB-classif* algorithm guarantees that the non-optimal images are presented only a negligible fraction of times ($\log N$ times out of a total budget N). The best action is thus sampled $N - O(\log N)$ times (this result will be demonstrated in the next section).

2.5 Adaptation of bandits to our problem

High probability bound for the classification error Let \mathcal{D} be a probability distribution in $\mathbb{R}^d \times \{0, 1\}$. Let \mathcal{H} be the set of binary linear classifiers in \mathbb{R}^d , i.e. if $(x, y) \sim \mathcal{D}$, (i.e. are drawn from \mathcal{D}) then $h(x)$ is the inferred class of the sample while the true class is y .

We define the $\{0, 1\}$ loss of a classifier h as

$$L_{\mathcal{D}}(h) = \mathbf{E}_{(x,y) \sim \mathcal{D}}(\mathbf{1}_{h(x) \neq y}).$$

Let h^* be the best linear classifier on \mathcal{D} for the $\{0, 1\}$ loss, i.e.

$$h^* = \arg \min_{h \in \mathcal{H}} L_{\mathcal{D}}(h).$$

Let now $\mathcal{X} = \{(x_1, y_1), \dots, (x_T, y_T)\}$ be T i.i.d. points in $\mathbb{R}^d \times \{0, 1\}$, sampled from \mathcal{D} .

We define the $\{0, 1\}$ empirical loss of a classifier h as

$$\hat{L}_{\mathcal{X}}(h) = \frac{1}{T} \sum_{t=1}^T (\mathbf{1}_{h(x_t) \neq y_t}).$$

Let $\widehat{h^*} \in \mathcal{H}$ be the best empirical classifier on \mathcal{X} in \mathcal{H} for the empirical $\{0, 1\}$ loss, i.e.

$$\widehat{h^*} = \arg \min_{h \in \mathcal{H}} \hat{L}_{\mathcal{X}}(h).$$

Theorem 1 (Vapnik, 1982) *We have with probability $1 - 2\delta$*

$$|L_{\mathcal{D}}(h^*) - \hat{L}_{\mathcal{X}}(\widehat{h^*})| \leq 2\sqrt{\frac{d(\log(2T/d) + 1) + \log(4/\delta)}{T}}.$$

In our setting and for task k , we have $1 - r_k^*$ which is the $\{0, 1\}$ loss of the best classifier for task k and $1 - \hat{r}_{k,t}$ which is the empirical $\{0, 1\}$ loss of the empirical best classifier for task k with $T_{k,t}$ samples.

We thus have with probability $1 - 2\delta$ for task k

$$|r_k^* - \hat{r}_{k,t}| \leq 2\sqrt{\frac{d(\log(2T_{k,t}/d) + 1) + \log(4/\delta)}{T_{k,t}}},$$

where d is the dimension of the feature space, which is in our case either 12 or 6 depending on whether we consider the rebound or not.

Overview of the way the algorithm works Let us now choose $\delta = 1/N$. As $T_{k,t} < N$, we have with probability $1 - 2/N$ that

$$|r_k^* - \hat{r}_{k,t}| \leq 2\sqrt{\frac{d(\log(2N/d) + 1) + \log(4N)}{T_{k,t}}} \leq \sqrt{\frac{a \log(N)}{T_{k,t}}},$$

where $a = 6(d + 1)$ when N is big enough.

We thus have with probability $1 - 2/N$

$$r_k^* \leq \hat{r}_{k,t} + \sqrt{\frac{a \log(N)}{T_{k,t}}} \leq r_k^* + 2\sqrt{\frac{a \log(N)}{T_{k,t}}}.$$

Note here that $B_{k,t} = \hat{r}_{k,t} + \sqrt{\frac{a \log(N)}{T_{k,t}}}$ is an upper bound with high probability on r_k^* .

On the event of large probability such that this is true for any k and any N , we know that we pull at time t a sub-optimal arm k if for the best arm $*$ with reward r^* , $B_{*,t} \leq B_{k,t}$, which according to the last equation leads to:

$$r^* \leq B_{*,t} \leq B_{k,t} \leq r_k^* + 2\sqrt{\frac{a \log(N)}{T_{k,t}}},$$

This means by a simple computation that we pull a sub-optimal arm k only if

$$T_{k,t} \leq 4 \frac{a \log(N)}{(r^* - r_k^*)^2}.$$

We then pull the suboptimal arms only a number of times in $O(\log(N))$, as $T_{k,N} \leq 4 \frac{a \log(N)}{(r^* - r_k^*)^2}$ and thus pull the optimal arm $N - O(\log(N))$ times, more precisely at least $N - \sum_{k \neq *} 4 \frac{a \log(N)}{(r^* - r_k^*)^2}$.

Finally, the error of the empirical classifier on the best arm is such that with high probability

$$|r^* - \hat{r}^*| \leq \sqrt{\frac{a \log(N)}{N - \sum_{k \neq *} 4 \frac{a \log(N)}{(r^* - r_k^*)^2}}}.$$

3 Results

3.1 Performances of the different tasks

Six right handed subjects did the experiment with real movements and three of them did an additional shorter experiment with imaginary movements. For four of the six subject the best performances for the real movement were achieved with the right hand, the two other subject's best tasks were the left hand and the feet.

Surprisingly, two of the subjects who did the imaginary experiment obtained better results while imagining moving their left hand than their right hand, which was the best task during the real movements experiment. For the last subject who did the imaginary experiment, the best task was the feet, as for the real movement experiment.

3.2 Influence of the use of the rebound

For the six subjects, we computed the classification rate of the best movement and compared the result with and without using the rebound features. For all subjects, using the rebound features increases the classification rate. It increases on average by 8.7% for real movements and by 5.9% for the imaginary movements (table 2).

3.3 Offline performances of the bandit algorithm

We compare the performance of the *UCB-classif* sampling strategy to a *uniform strategy*, i.e. the standard way of selecting a task, consisting of N/K presentations of each image.

movement	without rebound	with rebound	benefit
real	82%	91%	+8.7%
imaginary	82%	88%	+5.9%

Table 2: Off-line classification rate of the best movement with and without the beta rebound features. The increases are significant with $p > 95\%$.

One of the goals of our algorithm is to be able to select the best task among a large number of tasks. However, in our experiment, only a limited number of tasks were used (four), because we limited the length of the sessions in order not to tire the subjects, but still needed a sufficient number of samples for each task in order to be able to compute the classification rates.

To demonstrate the usefulness of our method for a larger number of tasks, we decided to create artificial (degraded) tasks by mixing the features of one of the real tasks (the feet) with different proportions of the features extracted during the resting period.

Movement	Number of presentations	Off-line classification
Right hand	28.6 ± 12.8	88.1%
Left hand	9.0 ± 7.5	80.5%
Feet	11.6 ± 9.5	82.6%
Tongue	4.5 ± 1.5	63.3%
Feet 80%	5.1 ± 2.6	71.4%
Feet 60%	4.0 ± 1.5	68.6%
Feet 40%	3.5 ± 1.0	59.2%
Feet 20%	3.5 ± 0.9	54.0%
Total presentations	70	

Table 3: Actions presented by the UCB-classif algorithm for subject 5 across the 500 simulated online experiments. Feet X% is a mixture of the features measured during feet movement and during the resting condition, with a X/100-X proportion. (The off-line classification rate of each action gives an idea of the performance of each action)

To obtain an realistic evaluation of the performance of our algorithm we use a bootstrap technique. More precisely, for each chosen budget N we simulate 500 online experiments by randomly selecting the data that will be used by the UCB-classif algorithm or the uniform strategy.

Table 3 shows, for one subject and for a fixed budget of $N = 70$, the average number of presentations of each task T_k , and its standard deviation, across the 500 simulated experiments. It also contains the off-line classification rate of each task to give an idea of the performances of the different tasks for this subject. We can see that very little budget is allocated to the Tongue movement and to the most degraded Feet tasks, which are the less discriminative actions, and that most of the budget is devoted to the Right hand, thus enabling a more efficient training.

Figure 4 and Table 4 show, for different budgets (N), the performance of the UCB-classif algorithm versus the uniform technique. The training of the classifier is done on the actions that would have been presented online and the testing on the remaining data. The classification results depend on which data

Budget N	experiment's length	Uniform	<i>UCB-classif</i>	Benefit
30	3min45	47.7%	64.4%	+16.7%
40	5min	58.5%	77.2%	+18.7%
50	6min15	63.4%	82.0%	+18.5%
60	7min30	67.0%	84.0%	+17.1%
70	8min45	70.1%	85.7%	+15.6%
100	12min30	77.6%	*	
150	18min45	83.2%	*	
180	22min30	85.2%	*	

Table 4: Comparison of the performances of the *UCB-classif* compared to the uniform strategy for different budgets, averaged over all subjects, for real movements. (The increases are significant with $p > 95\%$.) For each budget, we give an approximation of the length of the experiment (without the pauses) required to obtain this amount of data. For a budget $N > 70$ the *UCB-classif* could not be used for all the subjects because there was not enough data for the best action (One subject only underwent a session of 5 blocks and so only 50 samples of each motor task were recorded. If we try to simulate an on-line experiment using the *UCB-classif* with a budget higher than $N = 70$ it is likely to ask for a 51th presentation of the best task, which has not been recorded).

is used to simulate the online experiment. To give an idea of this variability, the first and last quartiles are plotted as error bars on the graphics.

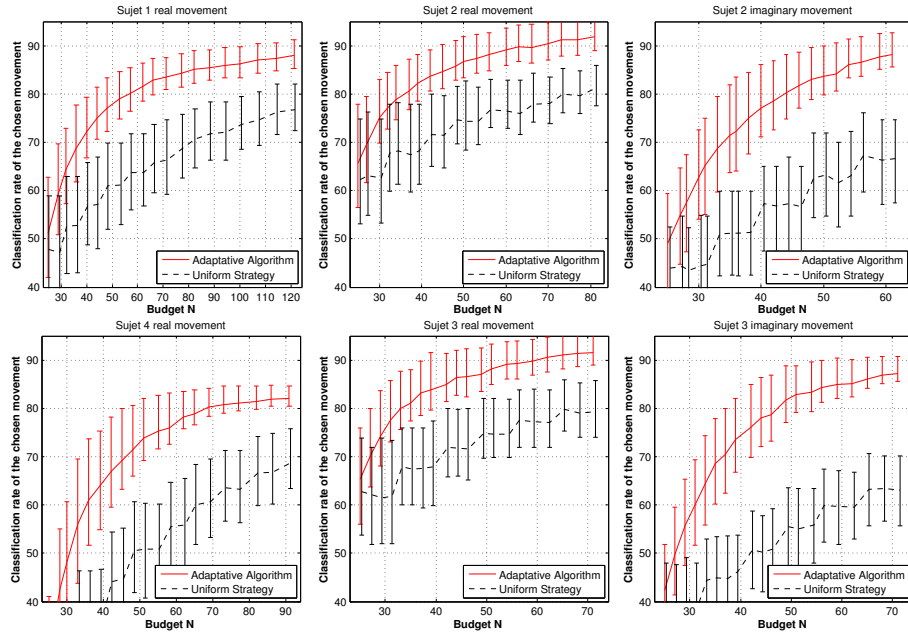


Figure 4: *UCB-classif* algorithm (full line, red) versus uniform strategy (dash line, black).

The UCB-classif strategy significantly outperforms the uniform strategy, even for relatively small N . On average on all the users it even gives better classification rates when using only half of the available samples, compared to the uniform strategy. Indeed, Table 4 shows that, to achieve a classification rate of 85% the *UCB-classif* only requires a budget of $N = 70$ whereas the uniform strategy needs $N = 180$.

4 Discussion and Perspectives

The results of the *UCB-classif* algorithm on this offline analysis are very promising, and we are preparing to investigate its performance during a real online experiment. Because it is not penalized by non-efficient tasks, this method should allow the exploration of a larger sets of motor tasks than customary up to now.

For this study we have chosen to use a very small set of fixed features (12 features, extracted from 3 electrodes, 2 frequency bands and 2 time-windows) that were calibrated on only one subject during a preliminary experiment. Although for all the subjects we obtained a classification accuracy for the best task superior to 85%, it is important to note that this accuracy would be yet improve by using a larger set of subject-specific features [15] and more advanced techniques (like the CSP [16] or feature selection [17]).

This is why, once the best task has been determined using *UCB-classif*, the acquired data should be used to automatically adjust the features for each subject. By rapidly focussing, during this exploratory phase, on the most discriminative task, sufficient data will be available to properly tune the features in addition to training the classifier.

An alternative strategy would be to start tuning the features during the use of *UCB-classif*. Unfortunately this would give rise to two issues: first, an important risk of over-fitting, especially for small amount of data, secondly a risk of favouring the tasks that have been the most sampled, and for which the features will thus be the best tuned.

The BCI targeted in this article use only one motor task to control a single brain-button. It would be even more interesting to aim for BCI that use two or more tasks to control more buttons. With this in view, we are studying an extended version of *UCB-classif* that will be able to select a pair (or a triplet) of tasks that will both maximize the classification rate of each task against non-movements and the classification rate between the two (or three) tasks.

Finally, our results also shed interesting light on the use of the beta rebound for BCI (Table 2): the rebound window has the disadvantage of increasing the latency of the response, but the advantage of improving the performances of the classification. For this reason, it could be interesting to use the rebound during the first stage of BCI operation, i.e. when the users are not very well trained and the classification results are not yet sufficient for a good control. After the users are better trained and sufficient data has been recorded to train the classifier, the rebound could be abandoned, to the benefit of a faster control.

5 Conclusion

The method presented in this paper falls in the category of adaptive BCI based on Machine Learning, but to our knowledge, it is the first one to deal with automatic task selection. *UCB-classif* is a new adaptive algorithm that allows to automatically select a motor task in view of an asynchronous brain-controlled button. By rapidly eliminating non-efficient motor tasks and focusing on the most promising ones, it enables a better task selection than a uniform strategy. Moreover, by more frequently presenting the selected task it allows a good training of the classifier. This algorithm enables to shorten the training period, or equivalently, to allow for a larger set of possible movements among which to select the best.

Acknowledgements

This work was partially supported by the French National Research Agency grant Co-Adapt.

References

References

- [1] Dennis J McFarland, William A Sarnacki, and Jonathan R Wolpaw. Electroencephalographic (EEG) control of three-dimensional movement. *Journal of Neural Engineering*, 7(3):036007, 2010.
- [2] Teodoro Solis-Escalante, Gernot Muller-Putz, Clemens Brunner, Vera Kaiser, and Gert Pfurtscheller. Analysis of sensorimotor rhythms for the implementation of a brain switch for healthy subjects. *Biomedical Signal Processing and Control*, 5(1):15 – 20, 2010.
- [3] Benjamin Blankertz, Guido Dornhege, Matthias Krauledat, Klaus-Robert Muller, and Gabriel Curio. The non-invasive berlin brain-computer interface: Fast acquisition of effective performance in untrained subjects. *NeuroImage*, 37(2):539 – 550, 2007.
- [4] G. Pfurtscheller and F. H. Lopes da Silva. Event-related EEG/MEG synchronization and desynchronization: basic principles. *Clinical Neurophysiology*, 110(11):1842 – 1857, 1999.
- [5] Gert Pfurtscheller and Christa Neuper. Motor imagery activates primary sensorimotor area in humans. *Neuroscience Letters*, 239(2-3):65 – 68, 1997.
- [6] Herbert Jasper and Wilder Penfield. Electrocorticograms in man: Effect of voluntary movement upon the electrical activity of the precentral gyrus. *European Archives of Psychiatry and Clinical Neuroscience*, 183:163–174, 1949. 10.1007/BF01062488.
- [7] Joan Fruitet, Maureen Clerc, and Thodore Papadopoulos. Preliminary study for an hybrid BCI using sensorimotor rhythms and beta rebound. In *International Journal of Bioelectromagnetism*, 2011.

- [8] M.-C. Dobrea and D.M. Dobrea. The selection of proper discriminative cognitive tasks - a necessary prerequisite in high-quality BCI applications. In *Applied Sciences in Biomedical and Communication Technologies, 2009. ISABEL 2009. 2nd International Symposium on*, pages 1 –6, 2009.
- [9] Jonathan R. Wolpaw, Niels Birbaumer, Dennis J. McFarland, Gert Pfurtscheller, and Theresa M. Vaughan. Brain-computer interfaces for communication and control. *Clinical Neurophysiology*, 113(6):767 – 791, 2002.
- [10] José del R. Millán, Frédéric Renkens, Josep Mourino, and Wulfram Gerstner. Brain-actuated interaction. *Artificial Intelligence*, 159(1-2):241 – 259, 2004.
- [11] Carmen Vidaurre and Benjamin Blankertz. Towards a cure for BCI illiteracy. *Brain Topography*, 23:194–198, 2010. 10.1007/s10548-009-0121-6.
- [12] H. Robbins. Some aspects of the sequential design of experiments. *Bulletin of the American Mathematics Society*, 58:527–535, 1952.
- [13] Yann Renard, Fabien Lotte, Guillaume Gibert, Marco Congedo, Emmanuel Maby, Vincent Delannoy, Olivier Bertrand, and Anatole Lécuyer. Open-ViBE: An open-source software platform to design, test, and use brain-computer interfaces in real and virtual environments. *Presence: Teleoperators and Virtual Environments*, 19(1):35–53, 2010.
- [14] P. Auer, N. Cesa-Bianchi, and P. Fischer. Finite time analysis of the multiarmed bandit problem. *Machine Learning*, 47(2-3):235–256, 2002.
- [15] M. Dobrea, D.M. Dobrea, and D. Alexa. Spectral EEG features and tasks selection process: Some considerations toward BCI applications. In *Multimedia Signal Processing (MMSP), 2010 IEEE International Workshop on*, pages 150 –155, 2010.
- [16] H. Ramoser, J. Muller-Gerking, and G. Pfurtscheller. Optimal spatial filtering of single trial EEG during imagined hand movement. *Rehabilitation Engineering, IEEE Transactions on*, 8(4):441 –446, 2000.
- [17] Joan Fruitet, Dennis J. Mcfarland, and Jonathan R. Wolpaw. A comparison of regression techniques for a two-dimensional sensorimotor rhythm-based brain-computer interface. *Journal of Neural Engineering*, 7(1), 2010.



Centre de recherche INRIA Sophia Antipolis – Méditerranée
2004, route des Lucioles - BP 93 - 06902 Sophia Antipolis Cedex (France)

Centre de recherche INRIA Bordeaux – Sud Ouest : Domaine Universitaire - 351, cours de la Libération - 33405 Talence Cedex
Centre de recherche INRIA Grenoble – Rhône-Alpes : 655, avenue de l'Europe - 38334 Montbonnot Saint-Ismier
Centre de recherche INRIA Lille – Nord Europe : Parc Scientifique de la Haute Borne - 40, avenue Halley - 59650 Villeneuve d'Ascq
Centre de recherche INRIA Nancy – Grand Est : LORIA, Technopôle de Nancy-Brabois - Campus scientifique
615, rue du Jardin Botanique - BP 101 - 54602 Villers-lès-Nancy Cedex
Centre de recherche INRIA Paris – Rocquencourt : Domaine de Voluceau - Rocquencourt - BP 105 - 78153 Le Chesnay Cedex
Centre de recherche INRIA Rennes – Bretagne Atlantique : IRISA, Campus universitaire de Beaulieu - 35042 Rennes Cedex
Centre de recherche INRIA Saclay – Île-de-France : Parc Orsay Université - ZAC des Vignes : 4, rue Jacques Monod - 91893 Orsay Cedex

Éditeur
INRIA - Domaine de Voluceau - Rocquencourt, BP 105 - 78153 Le Chesnay Cedex (France)
<http://www.inria.fr>
ISSN 0249-6399